WE CLAIM

1. A method of determining if the content of a source MPEG-2 stream, may be reduced, said method comprising the steps of:

a) examining said source stream to determine if a sequence_display_extension follows the most recent sequence header and sequence extension;

b) confirming that horizontal_size is greater than display_horizontal_size or that vertical_size is greater than display_vertical_size; and

c) if steps a) and b) are met, reducing the content of said source stream to create a reformatted stream.

2. The method of claim 1 where step c) comprises the steps of:

i) calculating the values of: width_mb and height_mb;

ii) calculating the values of: top, bottom, left and right;

iii) calculating the values: of top_mb, bottom_mb, left_mb and right_mb;

iv) substituting into said reformatted stream a portion of new_horizontal_size for horizontal_size and a portion of new_vertical_size for vertical_size; and

v) substituting into said reformatted stream a portion of new_horizontal_size for horizontal_size extension and a portion of new_vertical_size for vertical_size extension.

3. The method of claim 2 wherein the portion of step iv) is 12 bits.

4. The method of claim 2 wherein the portion of step v) is 2 bits.

5. The method of claim 2 further comprising the step of:

vi) removing picture_display_extension data from said source stream when creating said reformatted stream,

6. The method of claim 5 further comprising the step of:

vii) removing macroblocks from said source stream if their horizontal position is less than left_mb or greater than right_mb or if their vertical position is less than top_mb or greater than bottom_mb, when creating said reformatted stream

7. The method of claim 6 further comprising the step of:

viii) removing slices from said source stream which contain no macroblocks, when creating said reformatted stream

8. The method of claim 7 further comprising the step of:

ix) subtracting the value of top_mb from each slice_start_code, if slice_start_code becomes less than one, then setting it to one, in said reformatted stream.

9. A system for reducing the content of an MPEG-2 source stream, said system comprising:

a) a transcoder, said transcoder converting said source stream to a reformatted stream;

b) a transmitter connected to said transcoder, for transmitting said reformatted stream;

c) a receiver connected to said transmitter, for receiving said reformatted stream; and

d) a decoder connected to said receiver, for decoding said reformatted stream and providing as output a viewable stream.

10. The system of claim 9 wherein said transcoder comprises selector means to determine if said source stream may be reduced to create said reformatted stream, said logical means comprising the steps of:

a) examining said source stream to determine if a sequence_display_extension follows the most recent sequence header and sequence extension;

b) confirming that horizontal_size is greater than display_horizontal_size or that vertical_size is greater than display_vertical_size; and

c) if steps a) and b) are met, reducing the content of said source stream to create said reformatted stream.

11. The system of claim 10 wherein said transcoder further comprises calculation means for calculating and introducing into said reformatted stream new values for horizontal_size and vertical_size.

12. The system of claim 11 wherein said transcoder further comprises first removal means for removing picture_display_extension data from said source stream when creating said reformatted stream.

13. The system of claim 12 wherein said transcoder further comprises second removal means for removing macroblocks from said source stream when creating said reformatted stream.

14. The system of claim 13 wherein said transcoder further comprises slice removal means for removing slices from said source stream when creating said reformatted stream.

15. The system of claim 14 wherein said transcoder further comprises means for recalculating the value for slice_start_code fields.

16. A computer readable medium containing instructions for reducing the content of an MPEG-2 source stream, said instructions performing the steps of:

a) examining said source stream to determine if a sequence_display_extension follows the most recent sequence header and sequence extension;

b) confirming that horizontal_size is greater than display_horizontal_size or that vertical_size is greater than display_vertical_size; and

c) if steps a) and b) are met, reducing the content of said source stream to create a reformatted stream.

17. The medium of claim 16 wherein step c) comprises the steps of:

i) calculating the values of: width_mb and height_mb;

ii) calculating the values of: top, bottom, left and right;

iii) calculating the values of top_mb, bottom_mb, left_mb and right_mb;

iv) introducing into said reformatted stream a portion of new_horizontal_size for horizontal_size and a portion of new_vertical_size for vertical_size; and

v) introducing into said reformatted stream a portion of new_horizontal_size for horizontal_size extension and a portion of new_vertical_size for vertical_size extension.

18. The medium of claim 17 wherein the portion of step iv) is 12 bits.

19. The medium of claim 18 wherein the portion of step v) is 2 bits.

20. The medium of claim 19 further containing the step of:

vi) in creating said reformatted stream, removing picture_display_extension data from said source stream.

21. The medium of 20 further containing the step of:

vii) in creating said reformatted stream, removing macroblocks from said source stream if their horizontal position is less than left_mb or greater than right_mb or if their vertical position is less than top_mb or greater than bottom_mb.

22. The medium of claim 21 further containing the step of:

viii) in creating said reformatted stream, removing slices from said source stream which contain no macroblocks.

23. The medium of claim 22 further containing the step of:

ix) subtracting the value of top_mb from each slice_start_code, if slice_start_code becomes less than one, then setting it to one, in said reformatted stream.

24. A system for repositioning frames in an MPEG-2 stream said system comprising repositioning means, said repositioning means utilizing pan-scan information to relocate a display rectangle to a reconstructed frame.

25. The system of claim 24 wherein said repositioning means comprises:

a) selector means to determine if said frames in said stream are applicable for repositioning.;

b) first calculation means for calculating the values of: width_mb and height_mb;

c) second calculation means for calculating the values of: top, bottom, left and right;

d) third calculating means for calculating the values: of top_mb, bottom_mb, left_mb and right_mb; and

e) modification means for creating a reformatted stream based upon the input from steps b) to d).

26. The system of claim 25 wherein said modification means comprises picture removal means for removing picture_display_extension data from said source stream when creating said reformatted stream,

27. The system of claim 26 wherein said modification means further comprises macroblock removal means for removing macroblocks from said source stream if their horizontal position is less than left_mb or greater than right_mb or if their vertical position is less than top_mb or greater than bottom_mb, when creating said reformatted stream

28. The system of claim 27 wherein said modification means further comprises slice removal means for removing slices from said sourcestream which contain no macroblocks, when creating said reformatted stream

29. The system of claim 28 wherein said modification means further comprises subtraction means for subtracting the value of top_mb from each slice_start_code, if slice_start_code becomes less than one, then setting it to one, in said reformatted stream.

30. A video transcoder, said transcoder including a pan-scan module.

31. The transcoder of claim 30 wherein sad pan-scan module performs the following steps:

a) examining a source video stream to determine if a sequence_display_extension follows the most recent sequence header and sequence extension;

b) confirming that horizontal_size is greater than display_horizontal_size or that vertical_size is greater than display_vertical_size; and

c) if steps a) and b) are met, reducing the content of said source stream to create a reformatted stream.

32. The transcoder of claim 31 where step c) comprises the steps of:

i) calculating the values of: width_mb and height_mb;

ii) calculating the values of: top, bottom, left and right;

iii) calculating the values: of top_mb, bottom_mb, left_mb and right_mb;

iv) substituting into said reformatted stream a portion of new_horizontal_size for horizontal_size and a portion of new_vertical_size for vertical_size; and

v) substituting into said reformatted stream a portion of new_horizontal_size for horizontal_size extension and a portion of new_vertical_size for vertical_size extension.

33. The transcoder of claim 32 wherein the portion of step iv) is 12 bits.

34. The transcoder of claim 32 wherein the portion of step v) is 2 bits.

35. The transcoder of claim 32 wherein said pan-scan module performs the additional step of:

vi) removing picture_display_extension data from said source stream when creating said reformatted stream,

36. The transcoder of claim 35 wherein said pan-scan module performs the additional step of:

vii) removing macroblocks from said source stream if their horizontal position is less than left_mb or greater than right_mb or if their vertical position is less than top_mb or greater than bottom_mb, when creating said reformatted stream

37. The transcoder of claim 36 wherein said pan-scan module performs the additional step of:

viii) removing slices from said source stream which contain no macroblocks, when creating said reformatted stream

38. The transcoder of claim 37 wherein said pan-scan module performs the additional step of:

ix) subtracting the value of top_mb from each slice_start_code, if slice_start_code becomes less than one, then setting it to one, in said reformatted stream.

39. A pan-scan module, said pan-scan module residing within a video transcoder, said module comprising:

a) a selector for examining a source stream to determine if said source stream may be reduced;

b) a first calculator accepting as input said source stream to calculate the values of: width_mb and height_mb;

c) a second calculator accepting as input said source stream for calculating the values of: top, bottom, left and right;

d) a third calculator accepting as input said source stream for calculating the values of top_mb, bottom_mb, left_mb and right_mb; and

e) a modifer taking as input the calculations performed by said first, second and third calculators to create a reformatted stream.

40. The module of claim 39 wherein said modifier performs the steps of:

i) introducing into said reformatted stream a portion of a portion of new_horizontal_size for horizontal_size and a portion of new_vertical_size for vertical_size; and

ii) substituting into said reformatted stream a portion of new_horizontal_size for horizontal_size extension and a portion of new_vertical_size for vertical_size extension.

41. The module of claim 40 wherein the portion of step i) is 12 bits.

42. The module of claim 41 wherein the portion of step ii) is 2 bits.

43. The module of claim 40 wherein said modifier performs the additional step of:

iii) removing picture_display_extension data from said source stream when creating said reformatted stream,

44. The module of claim 43 wherein said modifier performs the additional step of:

iv) removing macroblocks from said source stream if their horizontal position is less than left_mb or greater than right_mb or if their vertical position is less than top_mb or greater than bottom_mb, when creating said reformatted stream

45. The module of claim 44 wherein modifier performs the additional step of:

v) removing slices from said source stream which contain no macroblocks, when creating said reformatted stream

46. The module of claim 45 wherein said modifier performs the additional step of:

vi) subtracting the value of top_mb from each slice_start_code, if slice_start_code becomes less than one, then setting it to one, in said reformatted stream.